

Estimating multidimensional probability fields using the Field Estimator for Arbitrary Spaces (FiEstAS) with applications to Astrophysics

Yago Ascasibar

*Universidad Autónoma de Madrid
Dpto Física Teórica, Campus de Cantoblanco, Madrid E-28049, Spain*

Abstract

The Field Estimator for Arbitrary Spaces (FiEstAS) computes the continuous probability density field underlying a given discrete data sample in multiple, non-commensurate dimensions. The algorithm works by constructing a metric-independent tessellation of the data space based on a recursive binary splitting. Individual, data-driven bandwidths are assigned to each point, scaled so that a constant “mass” M_0 is enclosed. Kernel density estimation may then be performed for different kernel shapes, and a combination of balloon and sample point estimators is proposed as a compromise between resolution and variance. A bias correction is evaluated for the particular (yet common) case where the density is computed exactly at the locations of the data points rather than at an uncorrelated set of locations. By default, the algorithm combines a top-hat kernel with $M_0 = 2.0$ with the balloon estimator and applies the corresponding bias correction. These settings are shown to yield reasonable results for a simple test case, a two-dimensional ring, that illustrates the performance for oblique distributions, as well as for a six-dimensional Hernquist sphere, a fairly realistic model of the dynamical structure of stellar bulges in galaxies and dark matter haloes in cosmological N-body simulations. Results for different parameter settings are discussed in order to provide a guideline to select an optimal configuration in other cases. Source code is available upon request.

Keywords: Kernel density estimation, multivariate data analysis

Email address: `yago.ascasibar@uam.es` (Yago Ascasibar)

1. Introduction

Given a point process where the D -dimensional probability density field $f(\mathbf{x})$ is sampled by N random points \mathbf{X}_i , the goal of density estimation is to infer the continuous function $f(\mathbf{x})$ from the discrete set of \mathbf{X}_i . One of the most popular approaches to the problem is kernel density estimation, in which the field is estimated by

$$\hat{f}(\mathbf{x}) = \frac{1}{|\mathbf{H}|} \sum_{i=1}^N K(\mathbf{H}^{-1}(\mathbf{x} - \mathbf{X}_i)) \quad (1)$$

where the kernel $K(u)$ is an even function that integrates to unity, and the bandwidth \mathbf{H} is a $D \times D$ matrix that specifies the scale, shape, and orientation of the kernel. The choice of this matrix has been thoroughly discussed in different contexts, and extensive reviews exist in the literature [e.g. 1, 2].

The importance of density estimation cannot be overstressed. Quite often, one is directly interested in the density itself; the FIESTAS algorithm was originally developed [3] to evaluate the density of particles in the six-dimensional phase space of positions and velocities. Although the problem has recently arisen considerable interest [e.g. 4, 5, 6, 7], it is of course only an anecdotal example. Nevertheless, it illustrates the difficulty of defining a metric (and related concepts, such as neighbourhood) in the general, non-Euclidean case. Although distances can be trivially defined in both three-dimensional subspaces, it is not clear how positions and velocities should be combined in order to produce a meaningful six-dimensional distance. It can be shown that a global scaling will only be appropriate for a certain region of the phase space, but not for the whole system [see the discussion in 3, 7]. In other words, the metric must adapt to the *local* structure of the data in order to recover the underlying density field.

In terms of applications, density estimation can be helpful in data mining problems. Unsupervised classification may be performed by identifying independent clusters with local density maxima, with boundaries set by the saddle points. In supervised classification, one can compute the probability distribution for each group c in the training set, $f_c(\mathbf{x})$, from the N_c data points belonging to it. Applying Bayes' theorem, the probability that a new

datum \mathbf{x} belongs to class c is given by

$$p(c|\mathbf{x}) = \frac{\pi_c f_c(\mathbf{x})}{\sum_i \pi_i f_i(\mathbf{x})} \quad (2)$$

where π_c denotes the prior probability of each class, and the sum in the denominator runs over all classes.

This work discusses the implementation of kernel smoothing in the Field Estimator for Arbitrary Spaces (FiESTAS). The algorithm is fully described in Section 2, and the results of benchmark tests are presented in Section 3. The main conclusions are summarized in Section 4.

2. Description of the algorithm

FiESTAS provides, for a given dataset $\{\mathbf{X}_i\}_{i=1,N}$ in D dimensions, the value of $f(\mathbf{x})$ at any arbitrary point \mathbf{x} . The algorithm involves the following steps:

1. Tessellation of the D -dimensional space.
2. Assignment of bandwidths to every data point.
3. Estimation of $f(\mathbf{x})$.
4. Bias correction (if necessary).

Each of them is described below, along with the different options and parameters that apply in each case.

2.1. Tessellation

The first step of the algorithm is the division of the data space in cells containing exactly one point. An important issue is the absence of a well-defined metric, which greatly increases the range of applicability of the method. Rather than using distances between data points, FiESTAS recursively divides the space by means of a k -d tree, one dimension at a time, until there is only one point per leaf.

There are several criteria to select the dimension to split at each step. The original version of FiESTAS [3] was fine-tuned to estimate densities in phase space, and it used the information that both the position and velocity subspaces are Euclidean. Moreover, it was imposed that divisions should take place alternatively in each subspace. A significant improvement over this scheme, proposed by [8], is the selection of the dimension with lower Shannon

entropy. Such a choice results in more divisions along the dimensions that show more structure, and therefore it adapts better to the distribution of the data. A very similar scheme was implemented in [9] to use FIESTAS in the context of Monte Carlo numerical integration: when a tree node has to be split, a histogram with $B = 1 + \sqrt{N_{\text{node}}}$ bins is built for each dimension, from the minimum to the maximum value attained by the corresponding coordinate. The log-likelihood for the histogram counts n_b to arise from a Poissonian distribution is given by

$$L_d = \ln(N_{\text{node}}!) - N_{\text{node}} \ln(B) - \sum_{b=1}^B \ln(n_{bd}!) \quad (3)$$

where the indices $1 \leq d \leq D$ and $1 \leq b \leq B$ denote the dimension and the bin number, respectively, n_{bd} is the number of points in each bin, and N_{node} is the total number of points in the node. The dimension with smaller L is divided at the point $x_{\text{split}} = (x_l + x_r)/2$, where x_l is the maximum x of all points lying on the “left” side ($b \leq b_{\text{split}}$) and x_r is the minimum x of the points lying on the “right” ($b > b_{\text{split}}$) side. The bin $1 \leq b_{\text{split}} < B$ is chosen in order that the number of points on each side is as close as possible to $N_{\text{node}}/2$.

A crude estimate of the density can be obtained as the inverse of the cell volume. As shown in [3], this estimate is very noisy, and it dramatically underestimates the density of particles near the boundary of the system. This becomes a critical problem in many dimensions, because the fraction of points affected quickly approaches unity as D increases. A simple correction was applied in [3] to data points at the boundary of the hypercubical domain, and a scheme based on the mean interparticle separation was used in [8] to adjust the shape of every tree node. In the present version of FIESTAS, such a correction is not necessary.

2.2. Bandwidth assignment

In principle, one should compute the $D(D+1)/2$ independent coefficients of the bandwidth matrix \mathbf{H} that minimize the mean integrated square error. However, doing that for every single datapoint can be impractical for large samples, and a simpler prescription has been adopted.

First, the bandwidth matrices are constrained to be diagonal. Although this is far from optimal when the data are distributed obliquely with respect to the coordinate axes [see e.g. 10, 11, 12], there is a substantial gain in

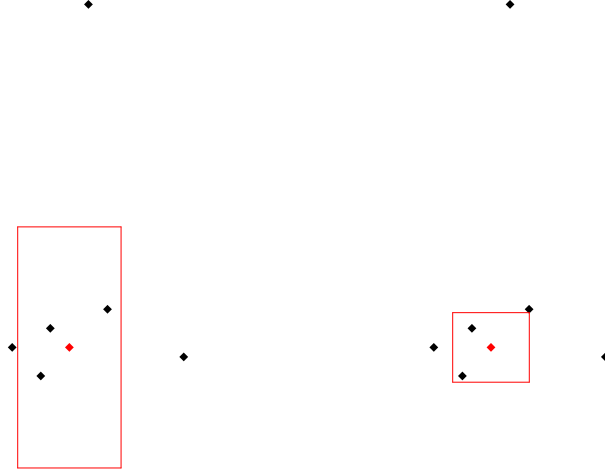


Figure 1: Bandwidth assignment for a given particle (plotted in red) in two dimensions. The box on the left panel represents $\mathbf{x} \pm \sigma$, where σ is the dispersion vector given by expression (4). The bandwidths (5) yield the box $\mathbf{x} \pm \mathbf{h}$ shown on the right panel, better adapted to the local distribution of data points.

speed, memory consumption, and code simplicity, by reducing the number of free parameters. This prescription will work well if the field is well sampled, although anisotropic kernels would perform better in oblique regions where the sampling is sparse.

The relation between the D smoothing lengths h_d of each point is estimated from the *local* dispersion of the data along each axis

$$\sigma_d^2 = \sum_{n=1}^{N_{\text{nei}}} X_{nd}^2 - \left(\sum_{n=1}^{N_{\text{nei}}} X_{nd} \right)^2 \quad (4)$$

where the index n refers to the N_{nei} neighbours defined by the FIESTAS tessellation. The smoothing lengths are then set to

$$h_d^2 = \frac{\sum_{n=1}^{N_{\text{nei}}} w_n X_{nd}^2}{\sum_{n=1}^{N_{\text{nei}}} w_n} - \left(\frac{\sum_{n=1}^{N_{\text{nei}}} w_n X_{nd}}{\sum_{n=1}^{N_{\text{nei}}} w_n} \right)^2 \quad (5)$$

with weights

$$w_n = \prod_{d=1}^D \frac{1}{\sigma_d} \exp \left[-\frac{(X_{nd} - X_{id})^2}{2\sigma_d^2} \right] \quad (6)$$

This measure is less sensitive to the presence of outliers than the simpler prescription $h_d = \sigma_d$ (see Figure 1).

In addition, FiESTAS offers the possibility of imposing a particular metric to any subspace by specifying a list of dimensions $\{d_l\}_{l=1,L}$ and the relative scale between them $\{s_l\}_{l=1,L}$. Defining $S = \prod_{l=1}^L s_l$ and $V = \prod_{l=1}^L h_{d_l}$,

$$h_{d_l} = s_l \frac{V}{S} \quad (7)$$

all other dimensions remaining unaltered. For instance, in phase space one could set dimensions $d_l = \{1, 2, 3\}$ (positions) to scale as $s_l = \{1.0, 1.0, 1.0\}$ and then impose the same Euclidean metric to the velocities, $d_l = \{4, 5, 6\}$. The relation between both spaces is not specified, and can vary freely from point to point.

Finally, the overall scale of the bandwidths is set so that the mass contained within the hypercube they define is equal to the user-defined parameter M_0 . The value of M_0 controls the degree of smoothing, and can be thought of as a constant (not necessarily integer) “number of neighbours” of the smoothing kernel. In order to compute it, each data point (of unit mass) is uniformly distributed over its cell, without any boundary correction,

$$m_i = \int_{\mathbf{x}_i - \mathbf{h}_i}^{\mathbf{x}_i + \mathbf{h}_i} \sum_{j=1}^N C_j(\mathbf{x}) \, d^D \mathbf{x} \quad (8)$$

where $C_j(\mathbf{x}) = 1$ if \mathbf{x} lies inside the j -th FiESTAS cell and 0 otherwise, and the bandwidths are scaled until $m_i = M_0$ within a 10 per cent tolerance. This is the only case in which the mass of the data is distributed like in the original implementation of FiESTAS.

2.3. Field estimation

At this point, it would be possible to estimate the density as

$$\hat{f}_K(\mathbf{x}) = \sum_{i=1}^N \prod_{d=1}^D \frac{1}{h_{id}} K\left(\frac{x_d - X_{id}}{h_{id}}\right) \quad (9)$$

where we have used a “product kernel” K . The current implementation includes top hat, $K(u) = 1/2$, triangular-shaped cloud, $K(u) = 1 - |u|$, and Epanechnikov, $K(u) = \frac{3}{4}(1 - u^2)$, kernels, where $-1 < u < 1$.

Apart from this possibility, FiESTAS can also combine $\hat{f}_K(\mathbf{x})$ with a top-hat balloon estimator

$$\hat{f}_B(\mathbf{x}) = \frac{1}{\prod_{d=1}^D 2\hat{h}_{Kd}(\mathbf{x})} \int_{\mathbf{x}-\hat{\mathbf{h}}_K(\mathbf{x})}^{\mathbf{x}+\hat{\mathbf{h}}_K(\mathbf{x})} \hat{f}_K(\mathbf{x}_0) \, d^D \mathbf{x}_0 \quad (10)$$

based on a local bandwidth

$$\hat{\mathbf{h}}_K(\mathbf{x}) = \frac{1}{\hat{f}_K(\mathbf{x})} \sum_{i=1}^N \mathbf{h}_i \prod_{d=1}^D \frac{1}{h_{id}} K\left(\frac{x_d - X_{id}}{h_{id}}\right) \quad (11)$$

interpolated from the individual particle bandwidths \mathbf{h}_i by using the same kernel as in equation (9).

2.4. Bias correction

In many, if not most, practical applications of the algorithm, one is interested in the value of the density field precisely at the locations of the sample points, and only $\hat{f}_i \equiv \hat{f}(\mathbf{X}_i)$ is evaluated. As discussed in [8], a positive bias that depends on the chosen kernel and its bandwidth arises in this particular case because we are not evaluating the density at a completely independent set of locations. The magnitude of this bias can be easily estimated for a uniform probability distribution by considering the average values of $\hat{f}_K(\mathbf{X}_i)$ and $\hat{f}_B(\mathbf{X}_i)$. In a uniform Poissonian distribution, $f(\mathbf{x}) = f_0$, all the smoothing lengths would be given by

$$M_0 \approx f_0(2h)^D \quad (12)$$

and thus

$$\langle \hat{f}_K(\mathbf{X}_i) \rangle = \prod_{d=1}^D \frac{K(0)}{h} + (N-1) \prod_{d=1}^D \frac{\langle K \rangle}{h} = \frac{[2K(0)]^D}{M_0} f_0 + \frac{N-1}{N} f_0 \quad (13)$$

whereas, for the balloon estimator,

$$\langle \hat{f}_B(\mathbf{X}_i) \rangle = \prod_{d=1}^D \frac{1}{2h} + (N-1) \prod_{d=1}^D \frac{\langle \int_{\mathbf{X}_i-h}^{\mathbf{X}_i+h} K \rangle}{h} = \frac{1}{M_0} f_0 + \frac{N-1}{N} f_0 \quad (14)$$

Therefore, assuming $N \gg 1$, the algorithm can apply a correction $\hat{f}_i = \hat{f}_i^{\text{uncorrected}} / (1+b)$ when only the \hat{f}_i are requested, where $b_K = [2K(0)]^D / M_0$ and $b_B = 1/M_0$. It is important to bear in mind that this correction factor

must *not* be applied in the general case, where the sample and evaluation points do not coincide. In particular, it should not be confused with the bias arising from the derivatives of f (note that, in fact, the values of b have been derived for a constant density), that has not been accounted for due to the difficulties associated to the estimation of local derivatives.

3. Results

The accuracy of the density reconstruction has been tested in two benchmark cases: a two-dimensional ring and a six-dimensional Hernquist sphere. We compare the performance of differnet kernels, as well as the scaling with the number N of sample points. Regarding the smoothing parameter, $M_0 = 2$ arguably represents a reasonable minimum, with smaller values yielding results (bandwidths and densities) that are dominated by the nearest data point. As will be shown below, increasing this parameter reduces the statistical variance of the estimator at the expense of resolution. A value $M_0 = 10$ is considered for reference, but higher values may be suitable depending on the user requirements, especially as the number of dimensions increases.

3.1. Two-dimensional ring

The first distribution is a ring in two dimensions with uniform density between an inner and an outer radius of 0.95 and 1.05, respectively, in arbitrary units. A random realization with 100 sample points is depicted in Figure 2, together with the density field returned by the FIESTAS algorithm under different parameter configurations. In all cases, the shape of the ring is correctly recovered, although some artifacts arise when the cells of the FIESTAS tessellation become extremely elongated. Since these artifacts are associated to individual points, they become more evident for large values of M_0 . As can be seen in the bottom panels, they are completely absent when a locally Euclidean metric (arguably the most appropriate for this problem, at least globally) is imposed.

The reconstruction obtained by the top-hat kernel has the obvious drawback of the sharp square edges, and the results obtained with the triangular-shaped cloud (not shown) or the Epanechnikov kernel are much more satisfactory in that sense. For $N = 100$, the Epanechnikov kernel with $M_0 = 10$ tends to severely oversmooth the density distribution. When the metric is constrained to be locally Euclidean ($h_x = h_y$ at every point), the width of the ring is systematically overestimated, but the recovered shape is perfectly

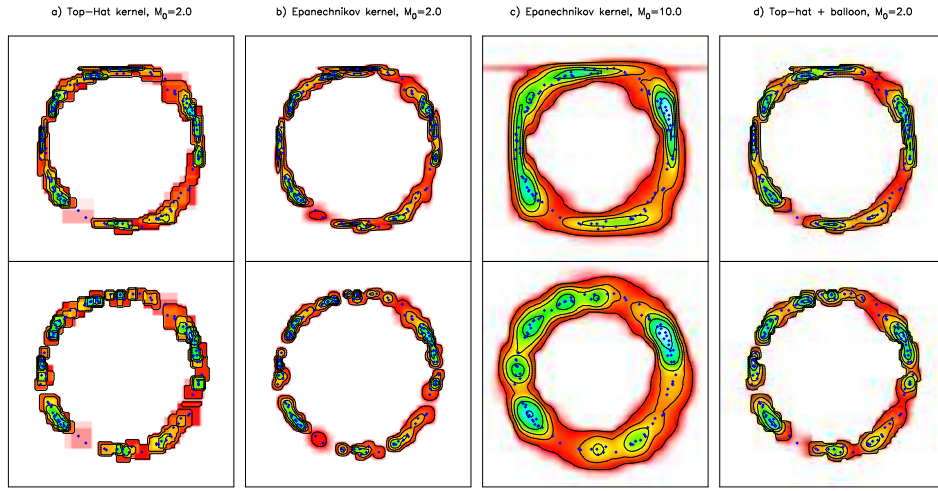


Figure 2: Density field recovered by the FiESTAS algorithm for a random realization of a two-dimensional ring distribution with 100 sample points (blue squares). Colours indicate local density, in arbitrary units, and contours enclose 5, 25, 50, 75, and 95 per cent of the mass. Dashed lines indicate the true distribution. The metric used on the top panels has not been constrained, whereas an Euclidean metric has been imposed on the bottom panels. Columns represent the results obtained for: a) top-hat kernel with $M_0 = 2$. b) Epanechnikov kernel with $M_0 = 2$. c) Epanechnikov kernel with $M_0 = 10$. d) FiESTAS balloon estimator, equation (10), combined with a top-hat kernel with $M_0 = 2$.

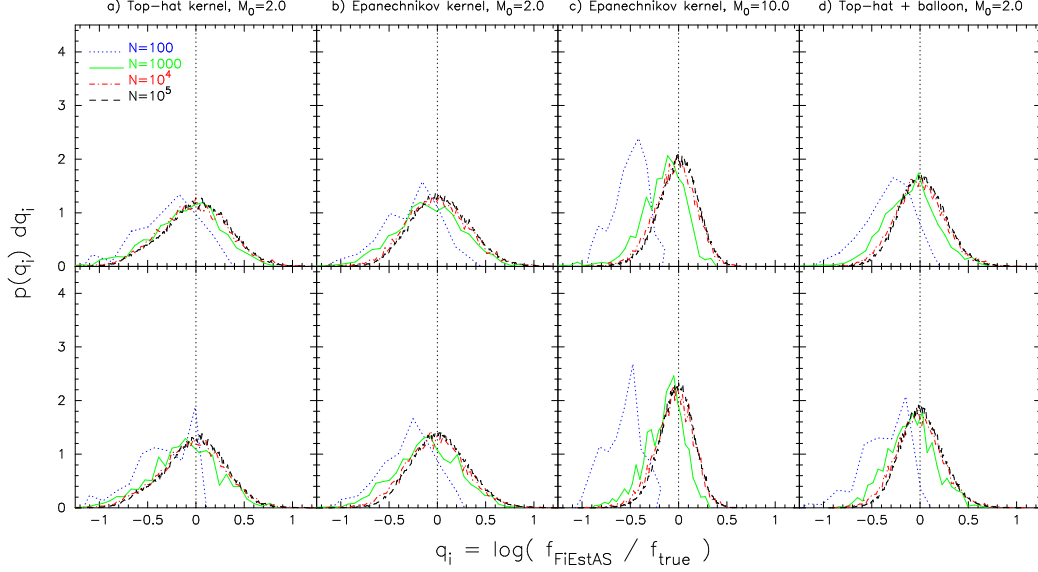


Figure 3: Probability distribution of the variable $q_i = \log \frac{\hat{f}(\mathbf{X}_i)}{f(\mathbf{X}_i)}$ for random realizations of the two-dimensional ring distribution with $N = 100, 1000, 10^4$ and 10^5 sample points. Columns represent different estimators, and an Euclidean metric has been imposed on the bottom panels.

circular. For the unrestricted metric, the density distribution is deformed into a slightly square shape aligned with the coordinate axes. This is due to the combined effect of the hypercubical FIESTAS tessellation (see [7] for a comparison of different schemes) and the diagonal bandwidth matrix. As a result, kernel shapes in “horizontal” or “vertical” regions tend to be more elongated, whereas $h_x \sim h_y$ in the “diagonal” regions, causing the “diamond” and “square” shapes observed for the inner and outer boundaries of the distribution. As stated above, it is in these oblique regions, poorly sampled within a smoothing volume, where an anisotropic kernel would certainly provide a significant advantage. Finally, combining a top-hat kernel with $M_0 = 2$ with the balloon estimator (10) yields a density field that is bracketed by the results of the Epanechnikov kernel with $M_0 = 2$ and $M_0 = 10$.

More quantitatively, the probability distribution of the variable $q_i = \log \frac{\hat{f}(\mathbf{X}_i)}{f(\mathbf{X}_i)}$ is shown in Figure 3 for several values of the number N of sample points between $N = 100$ and $N = 10^5$. The bias $\langle q_i \rangle$ and the variance $\sqrt{\langle q_i^2 \rangle - \langle q_i \rangle^2}$ of each estimator are quoted in Table 1. Since the density could already be properly reconstructed with $N \sim 1000$ points, the prob-

N	Top-hat	Epanechnikov	Epa., $M_0 = 10$	Top-hat+balloon
100	-0.28 ± 0.33	-0.27 ± 0.31	-0.50 ± 0.18	-0.32 ± 0.26
1000	-0.10 ± 0.38	-0.09 ± 0.35	-0.17 ± 0.24	-0.11 ± 0.29
10^4	-0.03 ± 0.36	-0.00 ± 0.32	-0.04 ± 0.22	-0.01 ± 0.26
10^5	-0.00 ± 0.34	0.03 ± 0.30	-0.01 ± 0.21	0.02 ± 0.24
100	-0.33 ± 0.30	-0.30 ± 0.29	-0.57 ± 0.19	-0.36 ± 0.24
1000	-0.12 ± 0.35	-0.11 ± 0.34	-0.15 ± 0.21	-0.09 ± 0.25
10^4	-0.04 ± 0.34	-0.01 ± 0.31	-0.05 ± 0.20	-0.01 ± 0.25
10^5	-0.02 ± 0.32	0.02 ± 0.29	-0.03 ± 0.18	0.01 ± 0.23

Table 1: Average value $\langle q_i \rangle$ and dispersion $\sqrt{\langle q_i^2 \rangle - \langle q_i \rangle^2}$ of the variable $q_i = \log \frac{\hat{f}(\mathbf{x}_i)}{f(\mathbf{x}_i)}$ for the two-dimensional ring distribution. Columns show the number of sample points and the results of each estimator. Top and bottom rows correspond to the unrestricted and Euclidean metrics, respectively.

ability distribution of q_i for this two-dimensional problem does not change much with N , with the exception of the oversmoothing shown by all estimators for $N = 100$. The bias correction was of the order of 20 – 50 per cent (0.09 – 0.18 dex) in all cases but the Epanechnikov kernel with $M_0 = 2$, for which it was about a factor of two. The variance also depends on the choice of a specific kernel and smoothing parameter M_0 , ranging from ~ 60 percent in the Epanechnikov kernel with $M_0 = 10$ to more than a factor of two for the top-hat kernel. It may be argued, though, that some of this dispersion is indeed physical, in the sense that it reflects the Poisson fluctuations inherent to the random realization of the ideal uniform distribution. In other words, there really are several clumps in the point distribution, and they are clearly visible in Figure 2. If one is interested in the actual physical density of these regions, its value should be higher than in those others that happen to contain less points. If, on the other hand, one is interested in the probability density field from which the sample was drawn, some statistical criterion has to be devised in order to test whether the fluctuations correspond to real variations of the field or are simply due to Poisson noise.

3.2. Hernquist sphere

The performance of the algorithm has also been tested by recovering the density of a six-dimensional Hernquist sphere [13]. This distribution is often used to model the central bulges of galaxies, as well as their dark matter

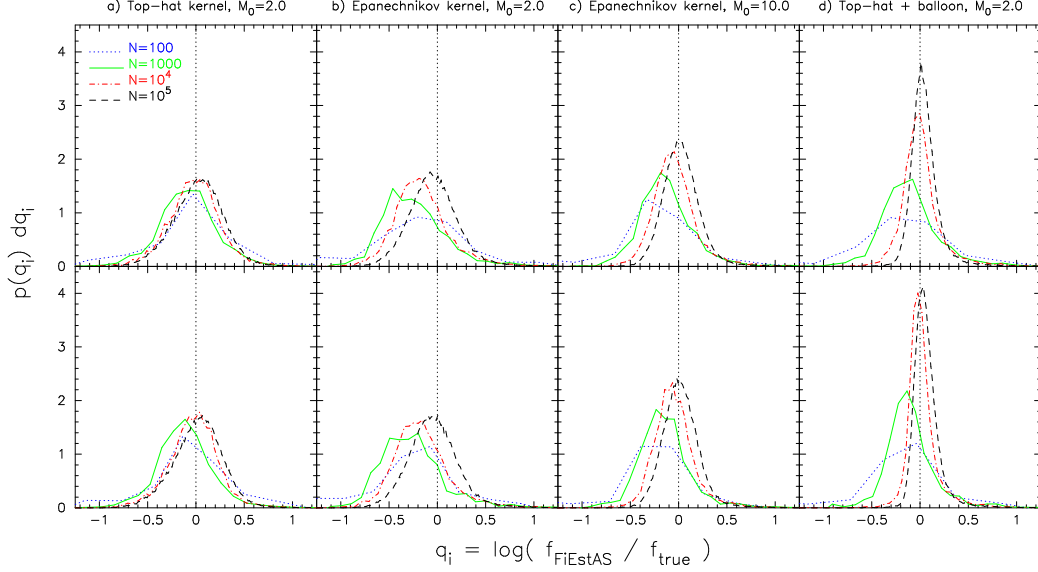


Figure 4: Probability distribution of the variable $q_i = \log \frac{\hat{f}(\mathbf{X}_i)}{f(\mathbf{X}_i)}$ for the six-dimensional Hernquist sphere. On the bottom panels, a three-dimensional Euclidean metric has been imposed locally to both the position and velocity subspaces.

haloes. The density of particles in the phase space of three-dimensional positions \mathbf{r} and velocities \mathbf{v} can be written as

$$f(\mathbf{r}, \mathbf{v}) = \frac{M/a^3}{4\pi^3 (2GM/a)^{3/2}} \frac{3 \sin^{-1} \sqrt{\epsilon} + \sqrt{\epsilon(1-\epsilon)}(1-2\epsilon)(8\epsilon^2 - 8\epsilon - 3)}{(1-\epsilon)^{5/2}} \quad (15)$$

in terms of the dimensionless specific binding energy of the particle

$$\epsilon = \frac{1}{1 + r/a} - \frac{v^2}{2GM/a} \quad (16)$$

and the total mass M and characteristic radius a of the system. The generation of a random realization of this distribution is described in [3].

Results obtained for different values of N are displayed in Figure 4 and Table 2. Overall, they are qualitatively similar to the example discussed in the previous section, with only minor differences due to the higher dimensionality of the problem and the very inhomogeneous nature of the Hernquist density distribution. In particular, the bias correction is much more important in six dimensions, reaching values as high as a factor of ~ 6.7 for the

N	Top-hat	Epanechnikov	Epa., $M_0 = 10$	Top-hat+balloon
100	-0.07 ± 0.46	-0.16 ± 0.49	-0.25 ± 0.49	-0.21 ± 0.56
1000	-0.08 ± 0.31	-0.24 ± 0.34	-0.13 ± 0.29	-0.11 ± 0.31
10^4	-0.01 ± 0.28	-0.16 ± 0.30	-0.04 ± 0.24	0.01 ± 0.22
10^5	0.03 ± 0.26	-0.04 ± 0.26	0.03 ± 0.20	0.05 ± 0.16
100	-0.10 ± 0.44	-0.21 ± 0.49	-0.28 ± 0.48	-0.24 ± 0.52
1000	-0.12 ± 0.29	-0.26 ± 0.33	-0.15 ± 0.28	-0.10 ± 0.27
10^4	-0.02 ± 0.26	-0.17 ± 0.30	-0.05 ± 0.23	0.02 ± 0.19
10^5	0.02 ± 0.26	-0.04 ± 0.26	0.02 ± 0.20	0.06 ± 0.14

Table 2: Average value $\langle q_i \rangle$ and dispersion $\sqrt{\langle q_i^2 \rangle - \langle q_i \rangle^2}$ of the variable $q_i = \log \frac{\hat{f}(\mathbf{X}_i)}{f(\mathbf{X}_i)}$ for the six-dimensional Hernquist sphere.

Epanechnikov kernel with $M_0 = 2$. Moreover, many more points are necessary in order to achieve an adequate sampling, and a clear evolution with N is now evident in the probability distribution of q_i . The negative bias observed at low N is mostly due to oversmoothing of the central regions, which contain the majority of the particles. The Hernquist distribution becomes optimally resolved for $N = 10^4 - 10^5$: the sampling within a smoothing volume becomes close to Poissonian, and the probability distribution of q_i approaches the asymptotic for the chosen kernel. As in the two-dimensional ring, the specification of a metric based on external knowledge of the problem (in this case, $h_1 = h_2 = h_3$ and $h_4 = h_5 = h_6$) affects the results only mildly.

4. Conclusions

Kernel density estimation has been implemented within the Field Estimator for Arbitrary Spaces (FIESTAS) algorithm, using different kernels and opening the possibility of combining sample point and balloon estimators. The only free parameters are the specific form of the kernel function (top-hat, triangular-shaped cloud and Epanechnikov kernels are provided by default) and the smoothing parameter M_0 . The bandwidth matrix, constrained to be diagonal, is automatically computed for every point. Additional constraints can be imposed by the user, but the test cases considered do not suggest that this results in a significant advantage. In fact, it has already been established for a wide range of cases [see e.g. 10] that independent bandwidths (arbitrary metric) do not lose power against the Euclidean metric, even if

the latter is true. A bias correction must be applied when one is only interested in the values of the density field exactly at the sample points \mathbf{X}_i . The magnitude of this correction depends on the details of the kernel, but it is already significant at $D = 2$ and tends to increase with dimensionality.

The optimal choice of kernel and smoothing parameter are, of course, problem-dependent. Based on the results presented in the previous section, the combination of a top-hat kernel with $M_0 = 2$ with the balloon estimator given by equation (10) seems to yield a reasonable compromise between accuracy (low dispersion) and resolution (small number of points required) for any number D of dimensions. This, however, may not hold in the general case, and the user is encouraged to experiment with different options. In particular, smaller values of the smoothing parameter M_0 are unlikely to provide useful results, but larger bandwidths may be helpful in order to reduce the statistical noise of the estimator at the expense of losing information about the small-scale structure of the data. The kernel shape has a much milder effect, but in some cases (e.g. if exact mass conservation is required), a sample point may be preferable to a balloon estimator. In this case, the Epanechnikov kernel is optimal for an L_2 loss criterion with fixed bandwidths [2], and this would be, in principle, the recommended choice.

Acknowledgments

Financial support for this work has been provided by the Spanish *Ministerio de Educación y Ciencia* (project AYA2007-67965-C03-03) and the European Science Foundation (ESF) for the activity entitled “Computational Astrophysics and Cosmology” (reference ASTROSIM 2027).

- [1] B. W. Silverman, Density estimation for statistics and data analysis, Monographs on Statistics and Applied Probability, London: Chapman and Hall, 1986, 1986.
- [2] M. P. Wand, M. C. Jones, Kernel Smoothing (Monographs on Statistics and Applied Probability), Chapman & Hall/CRC, 1995.
- [3] Y. Ascasibar, J. Binney, Numerical estimation of densities, MNRAS 356 (2005) 872–882. [arXiv:arXiv:astro-ph/0409233](#), [doi:10.1111/j.1365-2966.2004.08480.x](#).

- [4] M. Vogelsberger, S. D. M. White, A. Helmi, V. Springel, The fine-grained phase-space structure of cold dark matter haloes, *MNRAS* 385 (2008) 236–254. [arXiv:0711.1105](#), [doi:10.1111/j.1365-2966.2007.12746.x](#).
- [5] R. Wojtak, E. L. Lokas, G. A. Mamon, S. Gottlöber, A. Klypin, Y. Hoffman, The distribution function of dark matter in massive haloes, *MNRAS* 388 (2008) 815–828. [arXiv:0802.0429](#), [doi:10.1111/j.1365-2966.2008.13441.x](#).
- [6] I. M. Vass, M. Valluri, A. V. Kravtsov, S. Kazantzidis, Evolution of the dark matter phase-space density distributions of Λ CDM haloes, *MNRAS* 395 (2009) 1225–1236. [arXiv:0810.0277](#), [doi:10.1111/j.1365-2966.2009.14614.x](#).
- [7] M. Maciejewski, S. Colombi, C. Alard, F. Bouchet, C. Pichon, Phase-space structures - I. A comparison of 6D density estimators, *MNRAS* 393 (2009) 703–722. [arXiv:0810.0504](#), [doi:10.1111/j.1365-2966.2008.14121.x](#).
- [8] S. Sharma, M. Steinmetz, Multidimensional density estimation and phase-space structure of dark matter haloes, *MNRAS* 373 (2006) 1293–1307. [doi:10.1111/j.1365-2966.2006.11043.x](#).
- [9] Y. Ascasibar, FiEstAS sampling – a Monte Carlo algorithm for multidimensional numerical integration, *Computer Physics Communications* 179 (2008) 881–887. [arXiv:0807.4479](#), [doi:10.1016/j.cpc.2008.07.011](#).
- [10] M. P. Wand, M. C. Jones, Comparison of smoothing parameterizations in bivariate kernel density estimation, *Journal of the American Statistical Association* 88 (422) (1993) 520–528.
URL <http://www.jstor.org/stable/2290332>
- [11] T. Duong, M. L. Hazelton, Plug-in bandwidth selectors for bivariate kernel density estimation, *Journal of Nonparametric Statistics* 15 (2003) 17–30.
- [12] T. Duong, M. L. Hazelton, Cross-validation bandwidth matrices for multivariate kernel density estimation, *Scandinavian Journal of Statistics* 32 (3) (2005) 485–506.

- [13] L. Hernquist, An analytical model for spherical galaxies and bulges, *ApJ* 356 (1990) 359–364. doi:10.1086/168845.